The Role of Non-Ambiguous Words in Natural Language Disambiguation

Rada Mihalcea
Department of Computer Science and Engineering
University of North Texas
rada@cs.unt.edu

Abstract

This paper describes an unsupervised approach for natural language disambiguation, applicable to ambiguity problems where classes of equivalence can be defined over the set of words in a lexicon. Lexical knowledge is induced from non-ambiguous words via classes of equivalence, and enables the automatic generation of annotated corpora. The only requirements are a lexicon and a raw textual corpus. The method was tested on two natural language ambiguity tasks in several languages: part of speech tagging (English, Swedish, Chinese), and word sense disambiguation (English, Romanian). Classifiers trained on automatically constructed corpora were found to have a performance comparable with classifiers that learn from expensive manually annotated data.

1 Introduction

Ambiguity is inherent to human language. Successful solutions for automatic resolution of ambiguity in natural language often require large amounts of annotated data to achieve good levels of accuracy. While recent advances in Natural Language Processing (NLP) have brought significant improvements in the performance of NLP methods and algorithms, there has been relatively little progress on addressing the problem of obtaining annotated data required by some of the highest-performing algorithms. As a consequence, many of today's NLP applications experience severe data bottlenecks. According to recent studies (e.g. Banko and Brill 2001), the NLP research community should *"direct efforts towards increasing the size of annotated data collections"*, since large amounts of annotated data are likely to significantly impact the performance of current algorithms.

For instance, supervised part of speech tagging on English requires about 3 million words, each of them annotated with their corresponding part of speech, to achieve a performance in the range of 94-96%. State-of-the-art in syntactic parsing in English is close to 88-89% (Collins 96), obtained by training parser models on a corpus of about 600,000 words, manually parsed within the Penn Treebank project, an annotation effort that required 2 man-years of work (Marcus *et al.* 93). Increased level of problem complexity results in increasingly severe data bottlenecks. The data created so far for supervised English sense disambiguation consist of tagged examples for about 200 ambiguous words. At a throughput of one tagged example per minute (Edmonds 00), with a requirement of about 500 tagged examples per word (Ng & Lee 96), and with 20,000 ambiguous words in the common English vocabulary, this leads to about 160,000 hours of tagging – nothing less but 80 man-years of human annotation work. Information extraction, anaphora resolution, and other tasks also strongly require large annotated corpora, which often are not available, or can be found only in limited quantities.

Moreover, problems related to lack of annotated data multiply by an order of magnitude when languages other than English are considered. The study of a new language (according to a recent article in the Scientific American (Gibbs 02), there are 7,200 different languages spoken worldwide) implies a similar amount of work in creating annotated corpora required by the supervised applications in the new language.

In this paper, we describe a framework for unsupervised corpus annotation, applicable to ambiguity problems where classes of equivalence can be defined over the set of words in a lexicon. Part of speech tagging, word sense disambiguation, named entity disambiguation, are examples of such applications, where the same tag can be assigned to a set of words. In part of speech tagging, for instance, an *equivalence class* can be represented by the set of words that have the same functionality (e.g. noun). In word sense disambiguation, equivalence classes are formed by words with similar meaning (synonyms). The only requirements for this algorithm are a lexicon that defines the possible tags that a word might have, which is often readily available or can be build with minimal human effort, and a large raw corpus.

The underlying idea is based on the distinction between *ambiguous* and *non-ambiguous* words, and the knowledge that can be induced from the latter to the former via classes of equivalence. When building lexically annotated corpora, the main problem is represented by the words that, according to a given lexicon, have more than one possible tag. These words are *ambiguous* for the specific NLP problem. For instance, *"work"* is morphologically ambiguous, since it can be either a noun or a verb, depending on the context where it occurs. Similarly, *"plant"* carries on a semantic ambiguity, having both meanings of *"factory"* or *"living organism"*. Nonetheless, there are also words that carry only one possible tag, which are *non-ambiguous* for the given NLP problem. Since there is only one possible tag that can be assigned, the annotation of *non-ambiguous* words can be accurately performed in an automatic fashion. Our method for unsupervised natural language disambiguation relies precisely on this latter type of words, and on the equivalence classes that can be defined among words with similar tags.

Shortly, for an *ambiguous* word *W*, an attempt is made to identify one or more *non-ambiguous* words *W'* in the same class of equivalence, so that *W'* can be annotated in an automatic fashion. Next, lexical knowledge is induced from the *non-ambiguous* words *W'* to the *ambiguous words W* using classes of equivalence. The knowledge induction step is performed using a learning mechanism, where the automatically partially tagged corpus is used for training to annotate new raw texts including instances of the ambiguous word *W*.

The paper is organized as follows. We first describe the main algorithms explored so far in semi-automatic construction of annotated corpora. Next, we present our unsupervised approach for building lexically annotated corpora, and show how knowledge can be induced from *non-ambiguous* words via classes of equivalence. The method is evaluated on two natural language disambiguation tasks in several languages: part of speech tagging for English, Swedish, and Chinese, and word sense disambiguation for English and Romanian.

2 Related Work

Semi-automatic methods for corpus annotation assume the availability of some labeled examples, which can be used to generate models for reliable annotation of new raw data.

2.1 Active Learning

To minimize the amount of human annotation effort required to construct a tagged corpus, the active learning methodology has the role of selecting for annotation only those examples that are the most informative. While active learning does not eliminate the need of human annotation effort, it reduces significantly the amount of annotated training examples required to achieve a certain level of performance.

According to (Dagan *et al.* 95), there are two main types of active learning. The first one uses memberships queries, in which the learner constructs examples and asks a user to label them. In natural language processing tasks, this approach is not always applicable, since it is hard and not always possible to construct meaningful unlabeled examples for training. Instead, a second type of active learning can be applied to these tasks, which is *selective sampling*. In this case, several classifiers examine the unlabeled data and identify only those examples that are the most informative, that is the examples where a certain level of disagreement is measured among the classifiers.

In natural language processing, active learning was successfully applied to part of speech tagging (Dagan *et al.* 95), text categorization (Liere & Tadepelli 97), semantic parsing and information extraction (Thompson *et al.* 99).

2.2 Co-training

Starting with a set of labeled data, co-training algorithms, introduced by (Blum & Mitchell 98), attempt to increase the amount of annotated data using some (large) amounts of unlabeled data. Shortly, co-training algorithms work by generating several classifiers trained on the input labeled data, which are then used to tag new unlabeled data. From this newly annotated data, the most confident predictions are sought, which are subsequently added to the set of labeled data. The process may continue for several iterations.

Co-training was applied to statistical parsing (Sarkar 01), reference resolution (Mueller *et al.* 02), part of speech tagging (Clark *et al.* 03), statistical machine translation (Callison-Burch 02), and others, and was generally found to bring improvement over the case when no additional unlabeled data are used. However, as noted in (Pierce & Cardie 01), co-training has some limitations: too little labeled data yield classifiers that are not accurate enough to sustain co-training, while too many labeled examples result in classifiers that are "too accurate", in the sense that only little improvement is achieved by using additional unlabeled data.

2.3 Self-training

While co-training (Blum & Mitchell 98) and iterative classifier construction (Yarowsky 95) have been long considered to be variations of the same algorithm, they are however fundamentally different (Abney 02). The algorithm proposed in (Yarowsky 95) starts with a set of labeled data (seeds), and builds a classifier, which is then applied on the set of unlabeled data. Only those instances that can be classified with a precision exceeding a certain minimum threshold are added to the labeled set. The classifier is then trained on the new set of labeled examples, and the process continues for several iterations.

As pointed out in (Abney 02), the main difference between co-training and iterative classifier construction consists in the independence assumptions underlying each of these algorithms: while the algorithm from (Yarowsky 95) relies on precision independence, the assumption made in co-training consists in view independence.

Our own experiments in semi-supervised generation of sense tagged data (Mihalcea 02) have shown that self-training can be successfully used to bootstrap relatively small sets of labeled examples into large sets of sense tagged data.

2.4 Counter-training

Counter-training was recently proposed as a form of bootstrapping for classification problems where learning is performed simultaneously for multiple categories, with the effect of steering the bootstrapping process from ambiguous instances. The approach was applied successfully in learning semantic lexicons (Thelen & Riloff 02), (Yangarber 03).

3 Equivalence Classes for Building Annotated Corpora

The method introduced in this paper relies on classes of equivalence defined among *ambiguous* and *non-ambiguous* words. The method assumes the availability of: (1) a lexicon that lists the possible tags a word might have, and (2) a large raw corpus. The algorithm consists of the following three main steps:

1. Given a set $\mathcal{T}$ of possible tags, and a lexicon $\mathcal{L}$ with words W_i, i=1,$|\mathcal{L}|$, each word W_i admitting the tags T_j, j=1,$|W_i|$, determine equivalence classes $\mathcal{C}_{T_j}$, j=1,$|\mathcal{T}|$ containing all words that admit the tag T_j.

2. Identify in the raw corpus all instances of words that belong to only one equivalence class. These are *non-ambiguous* words that represent the starting point for the annotation process. Each such *non-ambiguous* word is annotated with the corresponding tag from $\mathcal{T}$.

3. The partially annotated corpus from step 2 is used to learn the knowledge required to annotate ambiguous words. Equivalence relations defined by the classes of equivalence $\mathcal{C}_{T_j}$ are used to determine *ambiguous* words W_i that are equivalent to the already annotated words. A label is assigned to each such ambiguous word by applying the following steps:
   (a) Detect all classes of equivalence $\mathcal{C}_{T_j}$ that include the word W_i.
   (b) In the corpus obtained at step 2, find all examples that are annotated with one of the tags T_j.
   (c) Use the examples from the previous step to form a training set, and use it to classify the current ambiguous instance W_i.

For illustration, consider the process of assigning a part of speech label to the word *"work"*, which may assume one of the labels NN (noun) or VB (verb). We identify in the corpus all instances of words that were already annotated with one of these two labels. These instances constitute training examples, annotated with one of the classes NN or VB. A classifier is then trained on these examples, and used to automatically assign a label to the current ambiguous word *"work"*. The following sections detail on the type of features extracted from the context of a word to create training/test examples.

3.1 Examples of Equivalence Classes in Natural Language Disambiguation

Words can be grouped into various classes of equivalence, depending on the type of language ambiguity.

Part of Speech Tagging

A class of equivalence is constituted by words that have the same morphological functionality. The granularity of such classes may vary, depending on specific application requirements. Corpora can be annotated using coarse tag assignments, where an equivalence class is constructed for each coarse part of speech tag (verb, noun, adjective, adverb, and the other main close-class tags). Finer tag distinctions are also possible, where for instance the class of plural nouns is separated from the class of singular nouns. Examples of such fine grained classes of morphological equivalence are listed below:

$\mathcal{C}_{NN}$ = { cat, paper, work }

$\mathcal{C}_{NNS}$ = { men, papers }
$\mathcal{C}_{VB}$ = { work, be, create }
$\mathcal{C}_{VBZ}$ = { lists, works, is, causes }

Word Sense Disambiguation

Words with similar meaning are grouped in classes of semantic equivalence. Such classes can be derived from readily available semantic networks like WordNet (Miller 95) or EuroWordNet (Vossen 98). For languages that lack such resources, the synonymy relations can be induced using bilingual dictionaries (Nikolov & Petrova 00). The granularity of the equivalence classes may vary from near-synonymy, to large abstract classes (e.g. artifact, natural phenomenon, etc.) For instance, the following fine grained classes of semantic equivalence can be extracted from WordNet:

$\mathcal{C}_{car}$ = { car, auto, automobile, machine, motorcar}
$\mathcal{C}_{mother}$ = { mother, female parent }
$\mathcal{C}_{begin}$ = { begin, get, start out, start, set about, set out, commence}

Named entity tagging

Equivalence classes group together words that represent similar entities (e.g. organization, person, location, and others). A distinction is made between named entity recognition, which consists in labeling new unseen entities, and named entity disambiguation, where entities that allow for more than one possible tag (e.g. names that can represent a person or an organization) are annotated with the corresponding tag, depending on the context where they occur.

Starting with a lexicon that lists the possible tags for several entities, the algorithm introduced in this paper is able to annotate raw text, by doing a form of named entity disambiguation. A named entity recognizer can be then trained on this annotated corpus, and subsequently used to label new unseen instances.

4 Evaluation

The method was evaluated on two natural language ambiguity problems. The first one is a part of speech tagging task, where a corpus annotated with part of speech tags is automatically constructed. The annotation accuracy of a classifier trained on automatically labeled data is compared against a baseline that assigns by default the most frequent tag, and against the accuracy of the same classifier trained on manually labeled data.

The second task is a semantic ambiguity problem, where the corpus construction method is used to generate a sense tagged corpus, which is then used to train a word sense disambiguation algorithm. The performance is again compared against the baseline, which assumes by default the most frequent sense, and against the performance achieved by the same disambiguation algorithm, trained on manually labeled data.

The precisions obtained during both evaluations are comparable with their alternatives relying on manually annotated data, and exceed by a large margin the simple baseline that assigns to each word the most frequent tag. Note that this baseline represents in fact a supervised classification algorithm, since it relies on the assumption that frequency estimates are available for tagged words.

Experiments were performed on several languages. The part of speech corpus annotation task was tested on English, Swedish, and Chinese, the sense annotation task was tested on English and Romanian.

4.1 Part of Speech Tagging

The automatic annotation of a raw corpus with part of speech tags proceeds as follows. Given a lexicon that defines the possible morphological tags for each word, classes of equivalence are derived for each part of speech. Next, in the raw corpus, we identify and tag accordingly all the words that appear only in one equivalence class (i.e. *non-ambiguous* words). On average (as computed over several runs with various corpus sizes), about 75% of the words can be tagged at this stage. Using the equivalence classes, we identify ambiguous words in the corpus, which have one or more equivalent *non-ambiguous* words that were already tagged in the previous stage. Each occurrence of such *non-ambiguous* equivalents results in a training example. The training set derived in this way is used to classify the ambiguous instances.

For this task, a training example is formed using the following features: (1) two words to the left and one word to the right of the target word, and their corresponding parts of speech (if available, or "?" otherwise); (2) a flag indicating whether the current word starts with an uppercase letter; (3) a flag indicating whether the current word contains any digits; (4) the last three letters of the current word. For learning, we use a memory based classifier (Timbl (Daelemans *et al.* 01)).

For each ambiguous word W_i defined in the lexicon, we determine all the classes of equivalence $\mathcal{C}_{T_j}$ to which it belongs, and identify in the training set all the examples that are labeled with one of the tags

T_j. The classifier is then trained on these examples, and used to assign one of the labels T_j to the current instance of the ambiguous word W_i.

The unknown words (not defined in the lexicon) are labeled using a similar procedure, but this time assuming that the word may belong to any class of equivalence defined in the lexicon. Hence, the set of training examples is formed with all the examples derived from the partially annotated corpus.

The unsupervised part of speech annotation is evaluated in two ways. First, we compare the annotation accuracy with a simple baseline, that assigns by default the most frequent tag to each ambiguity class.

Second, we compare the accuracy of the unsupervised method with the performance of the same tagging method, but trained on manually labeled data. In all cases, we assume the availability of the same lexicon. Experiments and comparative evaluations are performed on English, Swedish, and Chinese.

4.1.1 Part of Speech Tagging for English

For the experiments on English, we use the Penn Treebank Wall Street Journal part of speech tagged texts. Section 60, consisting of about 22,000 tokens, is set aside as a test corpus; the rest is used as a source of text data for training. The training corpus is cleaned of all part of speech tags, resulting in a raw corpus of about 3 million words. To identify classes of equivalence, we use a fairly large lexicon consisting of about 100,000 words with their corresponding parts of speech.

Several runs are performed, where the size of the lexically annotated corpus varies from as few as 10,000 tokens, up to 3 million tokens. In all runs, for both unsupervised or supervised algorithms, we use the same lexicon of about 100,000 words.

Training corpus size	Evaluation on test set	
	Training corpus built	
	automatically	manually
0 (baseline)	88.37%	
10,000	92.17%	94.04%
100,000	92.78%	94.84%
500,000	93.31%	95.76%
1,000,000	93.31%	96.54%
3,000,000	93.52%	95.88%

Table 1: Corpus size, and precision on test set using automatically or manually tagged training data (English)

Table 1 lists results obtained for different training sizes. The table lists: the size of the training corpus, the part of speech tagging precision on the test data obtained with a classifier trained on (a) automatically labeled corpora, or (b) manually labeled corpora. For a 3 million words corpus, the classifier relying on manually annotated data outperforms the tagger trained on automatically constructed examples by 2.3%. There is practically no cost associated with the latter tagger, other than the requirement of obtaining a lexicon and a raw corpus, which eventually pays off for the slightly smaller performance.

4.1.2 Part of Speech Tagging for Swedish

For the Swedish part of speech tagging experiment, we use text collections ranging from 10,000 words up to to 1 million words. We use the SUC corpus (SUC02), and again a lexicon of about 100,000 words. The tagset is the one defined in SUC, and consists of 25 different tags.

As with the previous English based experiments, the corpus is cleaned of part of speech tags, and run through the automatic labeling procedure. Table 2 lists the results obtained using corpora of various sizes. The accuracy continues to grow as the size of the training corpus increases, suggesting that larger corpora are expected to lead to higher precisions.

Training corpus size	Evaluation on test set	
	Training corpus build	
	automatically	manually
0 (baseline)	83.07%	
10,000	87.28%	91.32%
100,000	88.43%	92.93%
500,000	89.20%	93.17%
1,000,000	90.02%	93.55%

Table 2: Corpus size, and precision on test set using automatically or manually tagged training data (Swedish)

4.1.3 Part of Speech Tagging for Chinese

For Chinese, we were able to identify only a fairly small lexicon of about 10,000 entries. Similarly, the only part of speech tagged corpus that we are aware of does not exceed 100,000 tokens (the Chinese Treebank (Xue *et al.* 02)). All the comparative evaluations of tagging accuracy are therefore performed on limited size corpora. Similar with the previous experiments, about 10% of the corpus was set aside for testing. The remaining corpus was cleaned of part of speech tags and automatically labeled. Training on 90,000 manually labeled tokens results in an accuracy of 87.5% on the test data. Using the same training corpus, but automatically labeled, leads to a performance on the same test corpus of 82.05%. In an-other experiment, we increase the corpus size to about 2 million words, using the segmented Chinese corpus made publicly available by (Hockenmaier & Brew 98). The corpus is then automatically labeled with part of speech tags, and used as additional training data, resulting in a precision of 87.05% on the same test set.

The conclusion drawn from these three experiments is that non-ambiguous words represent a useful source of knowledge for the task of part of speech tagging. The results are comparable with previously explored methods in unsupervised part of speech tagging: (Cutting *et al.* 92) and (Brill 95) report a precision of 95-96% for part of speech tagging for English, using unsupervised annotation, under the assumption that all words in the test set are known. Under a similar assumption (i.e. all words in the test set are included in the lexicon), the performance of our unsupervised approach raises to 95.2%.

4.2 Word Sense Disambiguation

The annotation method was also evaluated on a word sense disambiguation problem. Here, the equivalence classes consist of words that are semantically related. Such semantic relations are often readily encoded in semantic networks, e.g. WordNet or EuroWordNet, can be induced using bilingual dictionaries (Nikolov & Petrova 00).

First, one or more *non-ambiguous* equivalents are identified for each possible meaning of the ambiguous word considered. For instance, the noun *"plant"*, with the two meanings of *"living organism"* and *"manufacturing plant"*, has the monosemous equivalents *"flora"* and *"industrial plant"*.

Next, the monosemous equivalents are used to extract several examples from a raw textual corpus, which constitute training examples for the semantic annotation task. The feature set used for this task consists of a surrounding window of two words to the left and right of the target word, the verbs before and after the target word, the nouns before and after the target word, and sense specific keywords. Similar with the experiments on part of speech tagging, we use the Timbl memory based learner.

The performance obtained with the automatically tagged corpus is evaluated against: (1) a simple baseline, which assigns by default the most frequent sense (as determined from the training corpus); and (2) a supervised method that learns from manually annotated corpora (the performance of the supervised method is estimated through ten-fold cross validations)

Word	Train. corpus	Test size	Most freq. sense	Disambig. precision Training corpus auto.	Disambig. precision Training corpus manual
bass	107	107	90.65%	92.52%	90.65%
crane	200	95	74.73%	71.57%	81.05%
motion	200	200	70.14%	75.62%	88.05%
palm	200	200	71.14%	80.59%	87.06%
plant	200	188	54.36%	69.14%	76.59%
tank	100	200	62.69%	63.69%	76.61%
AVERAGE	184	171	70.61%	76.60%	83.35%

Table 3: Corpus size, disambiguation precision using most frequent sense, and using automatically or manually sense tagged data. Automatic corpus annotation is performed using knowledge induced from non-ambiguous equivalents.

4.2.1 Word Sense Disambiguation for English

For this task, *monosemous* equivalents for six ambiguous words were determined based on WordNet synsets. The raw corpus consists of the British National Corpus, with about 100 million words, containing news article, scientific reports, novels, and spoken transcripts. Each monosemous equivalent is used to extract several examples (consisting of 4 sentences surrounding the focus word), up to a maximum of 100 examples per word sense.

Table 3 lists the six ambiguous words, the size of the training corpus automatically generated, the precision obtained on the test set using: (1) a simple heuristic that assigns the most frequent sense to all instances; (2) the classifier trained on (2a) automatically generated corpora, or (2b) manually labeled data.[1]

The disambiguation accuracy clearly exceeds the baseline, even for such small amounts of annotated corpora. While previous results reported in the literature for these words are sometimes larger (e.g. (Yarowsky 95)), note that the size of our corpus is limited to at most 100 examples per word sense, to allow for a one-to-one comparison with supervised methods (as opposed to thousands of annotated examples). Moreover, to avoid the bias introduced by the "one sense per discourse" paradigm, the examples that were manually annotated were selected exclusively from individual BNC texts, and therefore the "one sense per discourse" heuristic did not help the annotation process.

[1]The manually annotated corpus for these words is available from http://www.cs.unt.edu/~rada/downloads.html

4.2.2 Word Sense Disambiguation for Romanian

Since a Romanian WordNet is not yet available, *monosemous* equivalents for five ambiguous words were hand-picked by a native speaker using a paper-based dictionary. The raw corpus consists of a collection of Romanian newspapers collected on the Web over a three years period (1999-2002). The monosemous equivalents are used to extract several examples, again with a surrounding window of 4 sentences. An interesting problem that occurred in this task is the presence of gender, which may influence the classification decision. To avoid possible miss-classifications due to gender mismatch, the native speaker was instructed to pick the monosemous equivalents such that they all have the same gender (which is not necessarily the gender of their equivalent *ambiguous* word).

Table 4 lists the five ambiguous words, their monosemous equivalents, the size of the training corpus automatically generated, and the precision obtained on the test set using the simple most frequent sense heuristic and the instance based classifier. Again, the classifier trained on the automatically labeled data exceeds by a large margin the simple heuristic that assigns the most frequent sense by default. Since the size of the test set created for these words is fairly small (50 examples or less for each word), the performance of a supervised method could not be estimated.

Word	Training size	Most freq. sense	Disambig. precision
volum (book/quantity)	200	52.85%	87.05%
galerie (museum/tunnel)	200	66.00%	80.00%
canal (channel/tube)	200	69.62%	95.47%
slujba (job/service)	67	58.8%	83.3%
vas (container/ship)	164	60.9%	91.3%
AVERAGE	166	61.63%	87.42%

Table 4: Corpus size, disambiguation precision using most frequent sense, and using automatically sense tagged data (Romanian)

5 Conclusion

This paper introduced a framework for unsupervised natural language disambiguation, applicable to ambiguity problems where classes of equivalence can be defined over the set of words in a lexicon. Lexical knowledge is induced from non-ambiguous words via classes of equivalence, and enables the automatic generation of annotated corpora. The only requirements are a dictionary and a raw textual corpus. The method was tested on two natural language ambiguity tasks, on several languages. In part of speech tagging, classifiers trained on automatically constructed training corpora performed at accuracies in the range of 88-94%, depending on training size, comparable with the performance of the same tagger when trained on manually labeled data. Similarly, in word sense disambiguation experiments, the algorithm succeeds in creating semantically annotated corpora, which enable good disambiguation accuracies. In future work, we plan to investigate the application of this algorithm to very, very large corpora (Banko & Brill 01), and evaluate the impact on disambiguation performance.

Acknowledgments

Thanks to Sofia Gustafson-Capková for making available the SUC corpus, and to Li Yang for his help with the manual sense annotations.

References

(Abney 02) S. Abney. Bootstrapping. In *Proceedings of the 40st Annual Meeting of the Association for Computational Linguistics ACL 2002*, pages 360–367, Philadelphia, PA, July 2002.

(Banko & Brill 01) M. Banko and E. Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting of the Association for Computational Lingusitics (ACL-2001)*, Toulouse, France, July 2001.

(Blum & Mitchell 98) A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, 1998.

(Brill 95) E. Brill. Unsupervised learning of disambiguation rules for part of speech tagging. In *Proceedings of the ACL Third Workshop on Very Large Corpora*, pages 1–13, Somerset, New Jersey, 1995.

(Callison-Burch 02) C. Callison-Burch. Co-training for statistical machine translation. Unpublished M.Sc. thesis, University of Edinburgh, 2002.

(Clark *et al.* 03) S. Clark, J. R. Curran, and M. Osborne. Bootstrapping pos taggers using unlabelled data. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 49–55. Edmonton, Canada, 2003.

(Collins 96) M. Collins. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th Annual Meeting of the ACL*, Santa Cruz, 1996.

(Cutting *et al.* 92) D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing ANLP-92*, 1992.

(Daelemans *et al.* 01) W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. Timbl: Tilburg memory based learner, version 4.0, reference guide. Technical report, University of Antwerp, 2001.

(Dagan *et al.* 95) I. Dagan, , and S.P. Engelson. Committee-based sampling for training probabilistic classifiers. In *International Conference on Machine Learning*, pages 150–157, 1995.

(Edmonds 00) P. Edmonds. Designing a task for Senseval-2, May 2000. Available online at http://www.itri.bton.ac.uk/events/senseval.

(Gibbs 02) W.W. Gibbs. Saving dying languages. *Scientific American*, pages 79–86, 2002.

(Hockenmaier & Brew 98) J. Hockenmaier and C. Brew. Error-driven segmentation of chinese. In *12th Pacific Conference on Language and Information*, pages 218–229, Singapore, 1998.

(Liere & Tadepelli 97) R. Liere and P. Tadepelli. Active learning with committees for text categorization. In *Proceedings of the 14th Conference of the American Association of Artificial Intelligence, AAAI-97*, pages 591–596, Providence, RI, 1997.

(Marcus *et al.* 93) M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz. Building a large annotated corpus of english: the Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

(Mihalcea 02) R. Mihalcea. Instance based learning with automatic feature selection applied to Word Sense Disambiguation. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING-ACL 2002)*, Taipei, Taiwan, August 2002.

(Miller 95) G. Miller. Wordnet: A lexical database. *Communication of the ACM*, 38(11):39–41, 1995.

(Mueller *et al.* 02) C. Mueller, S. Rapp, and M. Strube. Applying co-training to reference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-02)*, Philadelphia, July 2002.

(Ng & Lee 96) H.T. Ng and H.B. Lee. Integrating multiple knowledge sources to disambiguate word sense: An examplar-based approach. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL-96)*, Santa Cruz, 1996.

(Nikolov & Petrova 00) T. Nikolov and K. Petrova. Building and evaluating a core of bulgarian wordnet for nouns. In *Proceedings of the Workshop on Ontologies and Lexical Knowledhe Bases OntoLex-2000*, pages 95–105, 2000.

(Pierce & Cardie 01) D. Pierce and C. Cardie. Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP-2001)*, Pittsburgh, PA, 2001.

(Sarkar 01) A. Sarkar. Applying cotraining methods to statistical parsing. In *Proceedings of the North American Chapter of the Association for Compuatational Linguistics, NAACL 2001*, Pittsburg, June 2001.

(SUC02) Stockholm Umea Corpus, 2002. http://www.ling.su.se/staff/sofia/suc/suc.html.

(Thelen & Riloff 02) M. Thelen and E. Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, Philadelphia, June 2002.

(Thompson *et al.* 99) C. A. Thompson, M.E. Califf, and R.J. Mooney. Active learning for natural language parsing and information extraction. In *Proceedings of the 16th International Conference on Machine Learning*, pages 406–414, 1999.

(Vossen 98) P. Vossen. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, Dordrecht, 1998.

(Xue *et al.* 02) N. Xue, F. Chiou, and M. Palmer. Building a large-scale annotated chinese corpus. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING-ACL 2002)*, Taipei, Taiwan, August 2002.

(Yangarber 03) R. Yangarber. Counter-training in discovery of semantic patterns. In *Proceedings of the 41 Annual Meeting of the Association for Computational Linguistics (ACL-03)*, Sapporo, Japan, July 2003.

(Yarowsky 95) D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95)*, pages 189–196, Cambridge, MA, 1995 1995.